LanceOtron: a deep learning peak caller for ATAC-seq, ChIP-seq, and DNase-seq

Lance D. Hentges[1,2], Martin J. Sergeant[1,2], Damien J. Downes[2], Jim R. Hughes[1,2] & Stephen Taylor[1*]

[1]MRC WIMM Centre for Computational Biology, MRC Weatherall Institute of Molecular Medicine, University of Oxford, Oxford, UK. [2]MRC Molecular Haematology Unit, MRC Weatherall Institute of Molecular Medicine, University of Oxford, Oxford, UK.

[*]To whom correspondence should be addressed.

Abstract

Genomics technologies, such as ATAC-seq, ChIP-seq, and DNase-seq, have revolutionized molecular biology, generating a complete genome's worth of signal in a single assay. Coupled with the use of genome browsers, researchers can now see and identify important DNA encoded elements as peaks in an analog signal. Despite the ease with which humans can visually identify peaks, converting these signals into meaningful genome-wide peak calls from such massive datasets requires complex analytical techniques. Current methods use statistical frameworks to identify peaks as sites of significant signal enrichment, discounting that the analog data do not follow any archetypal distribution. Recent advances in artificial intelligence have shown great promise in image recognition, on par or exceeding human ability, providing an opportunity to reimagine and improve peak calling. We present an interactive and intuitive peak calling framework, LanceOtron, built around image recognition using a wide and deep neural network. We hand-labelled 499Mb of genomic data, built 5,000 models, and tested with over 100 unique users from labs around the world. In benchmarking open chromatin, transcription factor binding, and chromatin modification datasets, LanceOtron outperforms the long-standing, gold-standard peak caller MACS2 with its increased selectivity and near perfect sensitivity. Additionally, this command-line optional approach allows researchers to easily generate optimal peak-calls using only a web interface. Together, the enhanced performance, and usability of LanceOtron will improve the reliability and reproducibility of peak calls and subsequent data analysis. This tool highlights the general utility of applying machine learning to genomic data extraction and analysis.

Main

Gene regulation is central to variation observed amongst cell types and disease states, and studying it often requires locating sites of specific DNA-protein interactions; the experimental procedure of chromatin precipitation followed by high throughput sequencing (ChIP-seq) is the method of choice for finding these sites. Similarly, given the stark functional difference of heterochromatin and euchromatin, identifying the regions of open and closed chromatin is also crucial to the study of epigenetics. Two commonly used assays for quantifying genome accessibility are ATAC-seq and DNase-seq. Taken together, these three sequencing-based, chromatin profiling assays are some of the most important experiments used to uncover genomic regulatory mechanisms[1].

Data from ATAC, ChIP, and DNase-seq are processed in a similar fashion: sequenced DNA fragments are aligned to the genome, and areas enriched for these fragments are recorded. Increased fragment density at true-positive biological events are called “peaks", because of the characteristic pattern of fragments produced in these areas. Besides these regions, enrichment also occurs due to noise from experimental procedures[2] or mapping errors, which are especially common in areas of low complexity[3]. Creating algorithms that can distinguish peaks from enriched noise, and which are also robust across bench equipment, sequencing depth, diverse tissue types, and chromosomal structure has remained a challenge.

Numerous bioinformatic tools, called peak callers, have been developed to distinguish peaks from noise employing different strategies to various degrees of success[4]. Peaks are prioritised using statistical tests that compare signals from putative peaks to background, which is assumed to consist of noise generated randomly according to an archetypal distribution, such as Poisson[5]. However background signal is nonrandom[6], appearing at increased levels in areas of open chromatin[7], at sites with inherent sequence bias[8] and over regions of varying copy number. As such, statistical tests often suffer from high false positive rates, but also leave room for potential false negatives, with the ratios of false positives to false negatives depending on the parameters defined[9]. Statistical peak callers can be improved through the use of matched negative controls to calculate the level of background noise, increasing the time required and the costs of the experiment. While peak callers such as MACS2[5] do not strictly require negative control tracks, forgoing them may sacrifice performance[10].

To address the well-known problems of peak callers, analysis pipelines employing quality control steps are common. The Encyclopedia of DNA Elements (ENCODE) consortium hosts numerous chromatin profiling assay datasets[11], and as such has a robust set of guidelines which includes recommendations for input controls, sequencing depth, library complexity, and blacklist regions where mapping errors are more prone to occur[12]. Multiple replicates are encouraged, and procedures exist for combining peak calls for the most efficient reduction in error[13]. Although these extensive measures greatly improve peak calls, high-throughput visual inspection showed numerous erroneous peak calls remain[14].

The ability, or inability, to reproduce published results is a prevalent concern amongst researchers[15]. This is, in part, due to the unintentional misapplication of statistics[16]. Command line peak callers such as MACS2 are routinely used with default settings rather than optimised parameters; aside from metaplots, no high-throughput methods allow direct investigation to quality check statistically significant, and nonsignificant peaks. Instead the significant regions are uploaded, along with a coverage track, to a genome browser such as UCSC[17] or IGV[18], where sections of the genome can be manually scanned or specific loci inspected. These tools make anything beyond a cursory inspection tedious, but because of the propensity of the statistical tests to be flawed, thoroughly exploring and refining peak calls is an important, though often overlooked task.

While forming robust statistical algorithms remains a challenge, it is often possible to call peaks from visual inspection using a genome browser. Rye et al. measured peak caller performance by creating a dataset of visually-verified peak calls, and inadvertently measured the performance of the humans in the process[19]. They found that transcription factors motifs were recovered more often from the manually labelled peaks than from the peak callers. Amazingly they also found that 80% of the software's false positives could be detected even without an input track, because the human peak callers could identify that these regions "lacked the expected visual appearance of a typical ChIP-seq peak". And while classifying regions by eye is seemingly dependent on an individual, Hocking et al. demonstrated a high consistency across labellers when judging peaks[9]. Visual inspection can be a credible, albeit impractical method for peak calling at a human genome scale.

Deep learning neural networks have been extremely successful in a number of general pattern detection tasks, such as image classification and voice recognition[20]. These techniques are being applied in biology as well, especially in genomics where there is an overabundance of data available for analysis[21]. Tools such as DeepSea[22] and Bassett[23] take genomic sequence as input, and can predict regulatory genomic features with high accuracy. Proof of principle studies have also shown promise for applying these techniques to peak calling[9,24].

Here we present LanceOtron, an open-source peak caller with a deep learning neural network, designed to increase selectivity without sacrificing sensitivity. LanceOtron considers the patterns of the aligned sequence reads, and their enrichment levels, and returns a probability that a region is a true peak with signal arising from a biological event. The user-friendly webtool has comprehensive filtering capabilities, and visualizations and interactive charts are generated automatically. LanceOtron is freely available at https://lanceotron.molbiol.ox.ac.uk.

Results

LanceOtron: a deep learning based peak caller with embedded visualization tools

LanceOtron operates similarly to other peak callers, taking a coverage file as input and returning enriched regions with associated scores as output; three main modules are available depending on the analysis being carried out (**Fig. 1a**). Required for all modules is a coverage file, input as a bigwig track, which is both compact and readily visualized. With widely used peak callers such as MACS2, assessing the quality of results cannot be done directly, rather the user must upload their output to a genome browser. This is somewhat restrictive for judging the quality of a peak call, in that users are limited to scanning some genomic regions to see if their results are sensible. To address this LanceOtron is built on the powerful MLV genome visualization software[14], which allows users to sort and filter results, as well as make thumbnail images of their peaks (**Fig. 1b, Supplementary video 1 & 2**).

The core of LanceOtron's peak scoring algorithm is a customized wide and deep model[25]. First, local enrichment measurements are taken from the maximum number of overlapping reads in a peak compared to its surroundings - chromosome-wide as well as 10 kilobases (kb) to 100kb regions (in 10kb increments). These measurements are used in a logistic regression model, which produces an enrichment score. An additional 2kb of signal, centered on the peak, is also input into a convolutional neural network (CNN). The CNN uses the relationship between the number of overlapping reads at all 2,000 points, i.e. the shape, to determine if the region is a peak arising from a biological event or noise. Finally a multilayer perceptron combines the outputs from CNN and logistic regression model, as well as the 11 local enrichment measurements to produce the overall peak score (**Fig. 1c**).

LanceOtron can also use unsupervised machine learning techniques, PCA, t-SNE, and UMAP, to cluster peaks based on shape. This allows for rapid assessment of peak call quality. Even peak calls following the strictest guidelines may contain low quality peaks. In this example, LanceOtron was used to analyse data from the ENCODE experiment ENCSR391NPE (ChIP-seq analysis on H3K27ac binding in 22Rv1 cells, conducted in two biological replicates). Each replicate was peak called separately, and only regions present in both calls were carried forward to the final list of enriched regions. However upon inspection it is clear low quality peaks are present. Using LanceOtron's deep learning based scoring, clustering, and visualization tools, these low quality regions can be readily identified (**Fig. 1d**).

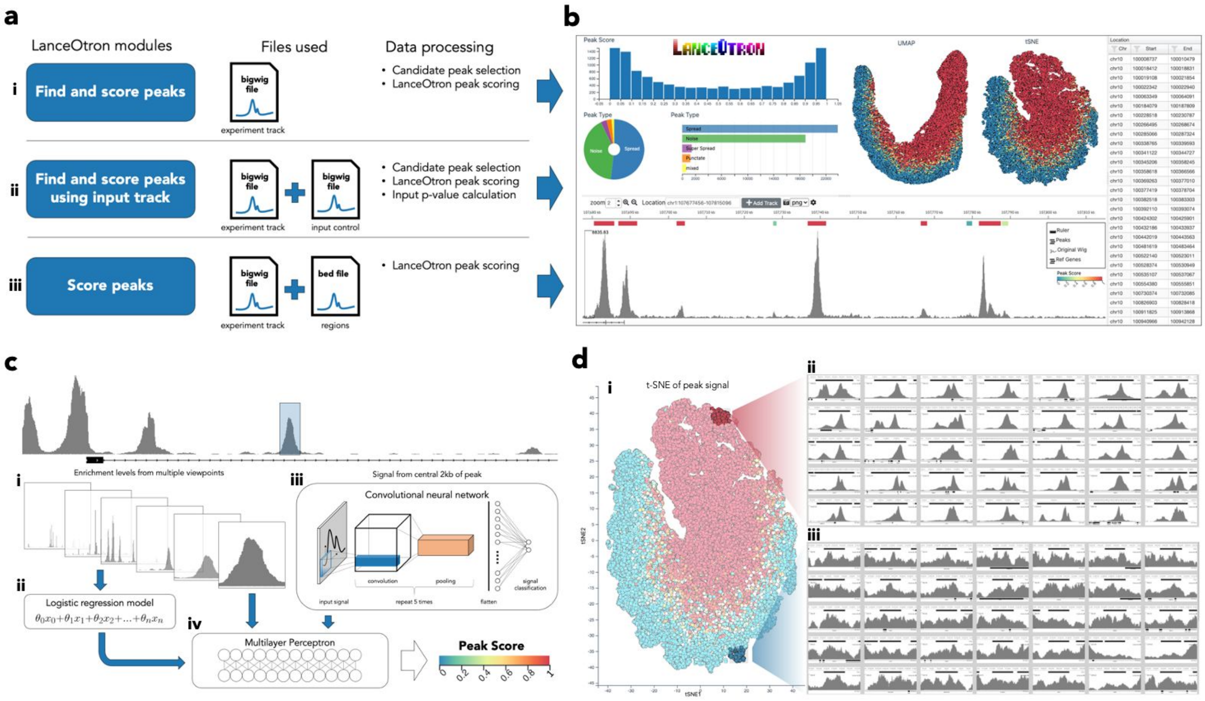


Fig. 1. LanceOtron, a deep learning based peak caller overview. a, Users can select from three different modules for making a peak call. **i,** Find and Score Peaks, determines the locations of enriched regions then scores them using LanceOtron's deep learning model. **ii,** Find and Score Peaks Using Input Track, additionally calculates p-values of enriched regions based on a separate input track. **iii,** Score Peaks, does not determine enriched regions, rather the genomic locations are uploaded as an additional file then scored. **b,** Peak call is visualized with linked and interactive bed file, charts, and genome browser. Filtering can be applied using LanceOtron's peak score, p-value, height, genomic coordinates, or any other criteria based on column in the interactive bed file. **c,** Overview of LanceOtron's neural network. i, Local enrichment is calculated from 10kb to 100kb regions in 10kb increments, chromosome-wide enrichment is also calculated. **ii,** The enrichment values are used as inputs for a logistic regression model. **iii,** Signal from the central 2 kilobases (kb) is fed into a convolutional neural network (CNN). **iv,** The output from the CNN, logistic regression model, and local enrichment values are all input into a multilayer perceptron, which produces the overall peak score for a given region. **d,** Regions found by ENCODE peak call, but scored with LanceOtron's model. **i,** peak calls are visualized using LanceOtron interactive t-SNE plot with thumbnails of regions selected on t-SNE plot for high **(ii)** and low **(iii)** LanceOtron peak scores.

Benchmarking LanceOtron

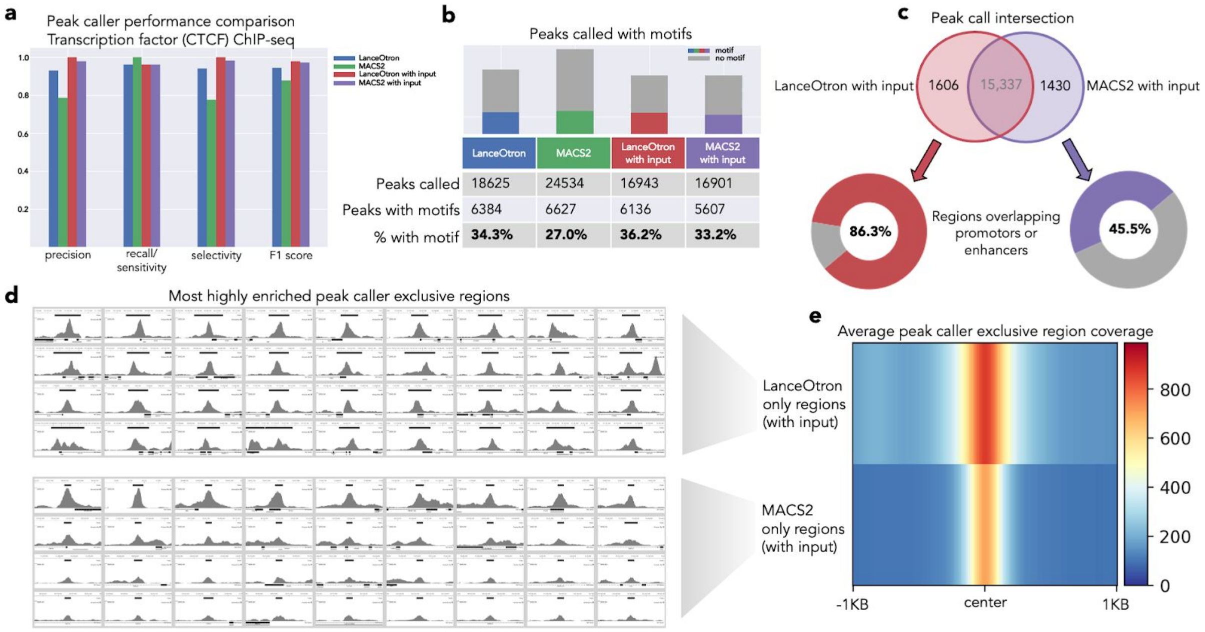
We benchmarked LanceOtron's performance with the ENCODE recommended peak caller MACS2, both using default settings (with and without an input control track when available). We compared peak calls from transcription factor ChIP-seq, histone ChIP-seq, and open chromatin assays (ATAC-seq and DNase-seq).

Transcription factor ChIP-seq

Our transcription factor dataset was CTCF in spleen primary cells, downloaded from ENCODE (ENCSR692ILH). We hand labelled 10 megabases (Mb) of the dataset, marking areas which were obviously peaks or noise (see methods) resulting in 109 annotations. Despite MACS2 without input having the best sensitivity, LanceOtron with input had the best overall accuracy (F1 score). This is attributed to LanceOtron's superior specificity while nearly matching MACS2's sensitivity (**Fig. 2a**).

While 10Mb is a considerable area to manually annotate, it represents a relatively small fraction of the human genome overall. To gain insight into the peak calls more generally, we performed motif analysis. The number of peaks called were similar between the different methods, though MACS2 (*without* input) was slightly higher: LanceOtron, 18,625; MACS2, 24,534; LanceOtron with input, 16,943; MACS2 with input, 16,901. Both peak calls from LanceOtron had the highest percentages of peaks which contained CTCF motifs: 36.2% with input, and 34.3% without. Percentage of the peak calls with motifs from MACS2 were 33.2% with input and 27.0% without (**Fig. 2b**).

LanceOtron with input had the highest F1 score when compared to MACS2 with input, and the differences between them were 1,606 peaks exclusively called with LanceOtron and 1,430 called with MACS2. Of these 86.3% of LanceOtron's peak calls overlapped with promoters or enhancers compared to just 45.5% of MACS2 only peak calls (**Fig. 2c**). When inspecting the regions called exclusively by MACS2, only a handful of the top enriched regions showed strong enrichment compared with LanceOtron (**Fig. 2d**). Further examining these exclusive peak calls, MACS2 regions were generally found in regions with less signal, and with peaks that were more narrow with lower enrichment than LanceOtron only peaks (**Fig. 2e**).



	LanceOtron	MACS2	LanceOtron with input	MACS2 with input
Peaks called	18625	24534	16943	16901
Peaks with motifs	6384	6627	6136	5607
% with motif	**34.3%**	**27.0%**	**36.2%**	**33.2%**

Fig 2. Benchmarking LanceOtron against MACS2 for peak calling transcription factor ChIP-seq. a, model performance metrics using labelled genomic regions of an ENCODE CTCF ChIP-seq dataset, 55 positive peaks and 67 noise regions. **b**, comparing the number of motifs contained the in peak calls generated from LanceOtron and MACS2. **c**, Venn diagram of peak calls from LanceOtron and MACS2. Regions which did not intersect assessed for overlap with promotors or enhancers. **d**, Selection of thumbnails from the most highly enriched regions called exclusively by either LanceOtron (top) or MACS2 (bottom). **e**, Average coverage of the regions called exclusively by either LanceOtron (top) or MACS2 (bottom).

Histone ChIP-seq

Our histone ChIP-seq datasets were H3K27ac in HAP-1 cells (ENCSR131DVD) and H3K4me3 in MG63 cells (ENCSR579SNM). For H3K27ac, LanceOtron correctly identified all 101 labelled regions (both with and without input), outperforming MACS2 (**Fig. 3a**). Performance was similar between peak callers in the H3K4me3 dataset, with MACS2 with input having slightly better sensitivity but LanceOtron with input having better precision, specificity, and F1 score (**Fig. 3b**).

To understand calls made by the peak callers more generally, we counted the number of transcription start sites (TSSs) overlapping with the returned regions. Due to the frequency with which TSSs are found in the genome, we restricted the analysis to the top 5,000 peaks called for each peak caller, and normalized the regions' size to 1kb. For H3K27ac, LanceOtron peaks overlapped with 27.5% more TSSs than MACS2, and 19.6% more when using input. We observed similar results for the H3K4me3 data, with LanceOtron peaks intersecting 15.0% more TSSs than MACS2, which increased to 60.3% with input (**Table 1**).

	LanceOtron	**MACS2**	**LanceOtron with input**	**MACS2 with input**
TSSs overlapping top H3K27ac peaks	10,847	8,505	10,906	9,115
TSSs overlapping top H3K4me3 peaks	12,885	11,202	13,054	8,142
Total ATAC peaks called	58,695	94,197		
% ATAC peaks in active regions (count)	15.0% (8,817)	7.6% (7,136)		
% ATAC peaks in inactive regions (count)	30.9% (18,149)	26.8% (25,198)		
Total DNase peaks called	16,719	67,461		
% DNase peaks in active regions (count)	17.6% (2,939)	7.1% (4,791)		
% DNase peaks in inactive regions (count)	36.9% (6,175)	26.5% (17,894)		

Table 1. LanceOtron and MACS2 peak call comparison for transcription start sites (TSSs) in histone ChIP-seq, and for active/inactive regions in open chromatin. Rows 1 and 2: counts of overlapping transcription start sites in top 5,000 peaks (highest q-value or peak score for LanceOtron and MACS2 respectively) from ENCODE H3K27ac dataset (ENCSR131DVD) in HAP-1 cells and H3K4me3 dataset (ENCSR579SNM) in MG63 cells. LanceOtron and MACS2 peak calls in GM12878 cells for ATAC-seq (ENCFF576DMC, rows 3-5) and DNase-seq (ENCSR000EMT, rows 6-8), showing total number of peaks found, and percentages with counts of regions found in active and inactive areas of the genome.

We also tested published datasets from Oh et al., who annotated peaks and noise for H3K27ac ChIP-seq in GM12878 cells and H3K4me3 in K562 cells[24]. Performance was consistent with our in-house labelled data, where MACS2 performed slightly better than LanceOtron on sensitivity, but LanceOtron besting MACS2 on precision, selectivity, and F1 score for both the H3K27ac data (**Fig. 3c**) and H3K4me3 data (**Fig. 3d**).

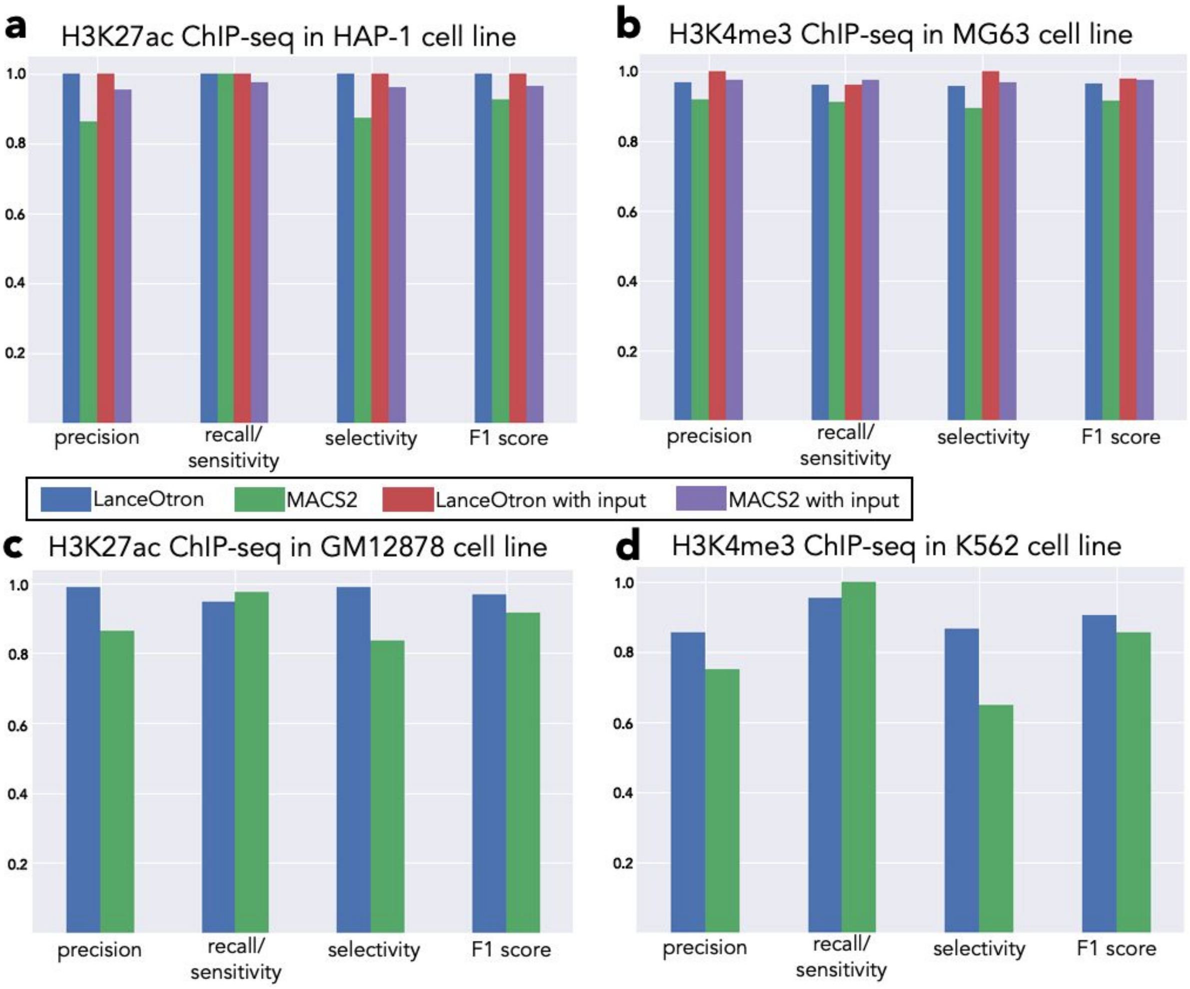


Fig. 3 Benchmarking LanceOtron against MACS2 for peak calling histone ChIP-seq. a, model performance metrics using labelled genomic regions of an ENCODE ChIP-seq datasets for H3K27ac in HAP-1 cell line, 45 positive peaks and 56 noise regions, and (**b**) H3K4me3 in MG63 cell line, 129 positive peaks and 95 noise regions. **c**, ChIP-seq dataset labelled by Oh et al. for H3K27ac in GM12878 cell line, 77 positive peaks and 73 noise regions, and (**d**) H3K4me3 in K562 cell line, 70 positive peaks and 66 noise regions.

ATAC and DNase-seq

In-house data for ATAC-seq consisted of 196 labelled regions in the cell line MCF-7 from ENCODE (ENCSR422SUG). LanceOtron matched MACS2 performance for recall/sensitivity, and surpassed it on precision, sensitivity, and F1 score (**Fig. 4a**).

Results were similar for our in-house DNase-seq data, consisting of 224 labelled regions in the cell line A549 from ENCODE (ENCSR000ELW). MACS2 outperformed LanceOtron for recall/sensitivity, but had a very high false positive rate. Consequently LanceOtron beat MACS2 on precision, sensitivity, and F1 score (**Fig. 4b**).

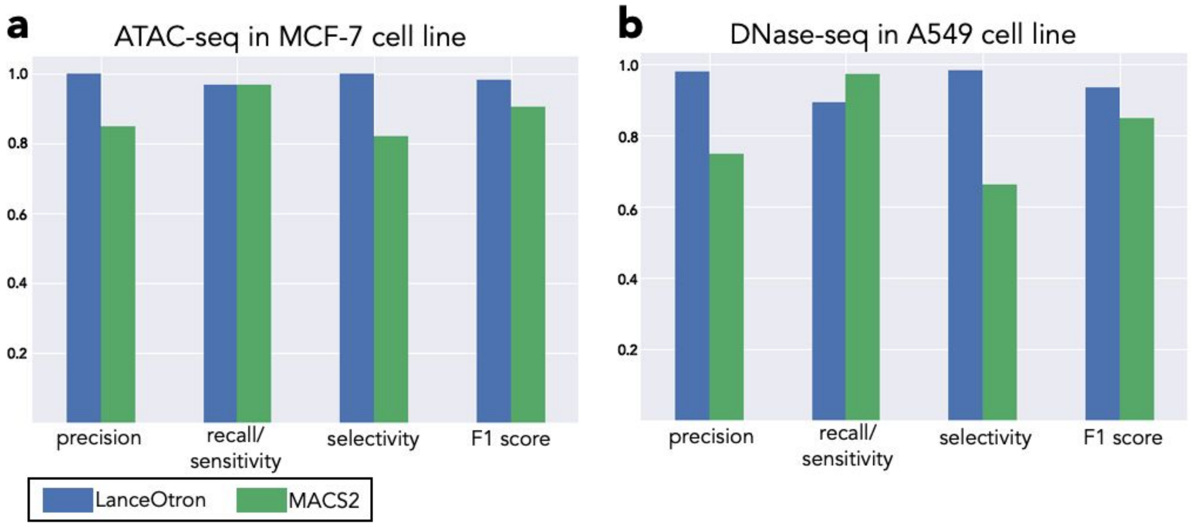


Fig. 4 Benchmarking LanceOtron against MACS2 for calling open chromatin. a, model performance metrics using labelled genomic regions of an ENCODE ATAC-seq dataset in MCF-7 cell line, 101 positive peaks and 95 noise regions, and (**b**) DNase-seq in A549 cell line, 114 positive peaks and 110 noise regions.

We also compared peak calling performance on GM12878 cells for ATAC (ENCFF576DMC) and DNase (ENCSR000EMT). Here we used published annotations from Tarbell and Liu[26], whereby they defined active and inactive (heterochromatin) areas of the genome using enhancer and promoter data with the software GenoSTAN[27].

For ATAC-seq peak calling on the annotated GM12878 data, both peak callers found a large number of peaks in heterochromatin: 18,149 and 25,198 regions for LanceOtron and MACS2 respectively. However LanceOtron had a larger percentage of peaks called in active areas of the genome, 15.0%, compared to MACS2 at 7.6%. Despite MACS2 calling 60% more peaks than LanceOtron, it found 1,681 fewer peaks in active areas of the genome. DNase performance with the annotated GM12878 followed the same general trend as ATAC-seq. Still a large number of peaks were found in heterochromatin: 6,175 for LanceOtron and 17,894 regions for MACS2. LanceOtron also had a larger percentage of peaks called in active areas of the genome (17.6%) compared to MACS2 (7.1%). MACS2 once again called many more peaks than LanceOtron, 50,742 additional regions, but only 3.6% of these were found in active areas of the genome (**Table 1**).

Discussion

LanceOtron is a deep learning based peak caller for genomic signal analysis, with a full user-friendly interface designed for interrogation of large datasets. Its CNN learns the shape of a region, and in combination with enrichment calculations, allows for more powerful analysis. Traditional peak callers return only those regions which cross a high statistical threshold. LanceOtron, however, returns all enriched regions above a relatively low threshold, along with their associated peak scores, p-values, etc. This makes LanceOtron akin to an automated annotation tool, returning a greater breadth of data about the experiment. It's function as a peak caller is achieved by LanceOtron's comprehensive filtering, further highlighting the importance of a powerful user interface.

Benchmarking transcription factor ChIP-seq data revealed that many of the unique regions found with LanceOtron were associated with enhancers or promoters compared with MACS2. Upon inspecting the DNase track for the cell type, it is clear many of the regions missed by MACS were in regions of open chromatin. These areas also had some increase in signal on the input track as expected[7], however not enough to be statistically significant as determined by LanceOtron's Poisson test. This could be due to the necessarily high p-value threshold set by MACS2 in order to better reduce false positives genome-wide, but at the cost of sensitivity in active regions of the genome.

LanceOtron had a comprehensive development process during which over 100 unique users have tested the tool, with over 30 users creating 10 projects or more. We have learned how labs around the world analyse their chromatin profiling assays, and we designed our workflow around this experience. One feature several groups have requested is the ability to peak call and compare multiple tracks simultaneously. As LanceOtron continues to develop we hope to bring this feature forward in future versions.

Our module lineup was also informed by user feedback. Originally, we developed the "Find and Score Peaks" module which used a bigwig track as its sole input. Our benchmarking shows that this module outperforms MACS2 and is on par with MACS2 with input, making this a good option when input is not available. The "Find and Score Peaks with Input" module builds on this, reducing false positives in areas of high signal due to increased noise. The "Score Peaks" module allows users to upload peak calls made by other tools. This means groups can easily add LanceOtron to their current workflow to score their peaks with its neural network, visualize, and filter their results. This module is also important for data reproducibility or peer reviewing data. Peak calls made by other groups can easily be uploaded, scored, and visualized - or if the peak call was made with LanceOtron, it can just be made public for easy review.

In summary LanceOtron is a powerful peak caller and analysis tool for ATAC-seq, ChIP-seq, and DNase-seq. Across a range of different datasets and data types, LanceOtron outperformed the industry-standard MACS2. It is designed to accommodate current workflows as a visualization, annotation, and filtering tool, or to be used further upstream as a peak caller leveraging a powerful deep learning neural network alongside traditional statistical tests.

Methods

Deep learning model

Training data

The data used to train the neural network was obtained from ENCODE. To generate a complete list of experiments which met our specifications we used ENCODE's REST API (scripts and outputs available on GitHub). We filtered the results to samples which were "released" status at the time of search inquiry, and aligned to human reference genome hg38 as BAM files; for H3K27ac, H3K4me3, and TF ChIP-seq experiments, the availability of a corresponding control track was also required. While infrequent, samples were excluded if ENCODE metadata did not include information on single-end versus paired-end sequencing. The number of samples meeting this criteria was 3902 (74 ATAC, 911 DNase, 305 H2K27ac, 463 H3K4me3, 2149 transcription factor samples). We sampled 10 paired-end datasets for each category at random from each experiment type, except in H3K4me3 experiments where only 6 samples available were paired-end, and so 4 single end experiments were included. This resulted in 38 unique biosample types, 9 unique transcription factor ChIP-seq targets plus 2 histone ChIP-seq targets (**Table 2**).

Experiment type			ENCODE ID numbers		
Assay	Target	Tissue	Experiment	BAM file	Control BAM
ATAC-seq	Open chromatin	Breast epithelium	ENCSR955JSO	ENCFF656OYT	
ATAC-seq	Open chromatin	Tibial artery	ENCSR630REB	ENCFF168OTV	
ATAC-seq	Open chromatin	Foreskin keratinocyte	ENCSR290YMN	ENCFF799HAR	
ATAC-seq	Open chromatin	Adrenal gland	ENCSR113MBR	ENCFF436NOT	
ATAC-seq	Open chromatin	Foreskin keratinocyte	ENCSR158XTU	ENCFF784DSJ	
ATAC-seq	Open chromatin	Foreskin keratinocyte	ENCSR677MJF	ENCFF764CQI	
ATAC-seq	Open chromatin	Transverse colon	ENCSR668VCT	ENCFF377DAO	
ATAC-seq	Open chromatin	Sigmoid colon	ENCSR548QCP	ENCFF482HAC	
ATAC-seq	Open chromatin	Tibial nerve	ENCSR831KAH	ENCFF277DNH	
ATAC-seq	Open chromatin	Thyroid gland	ENCFF710ELD	ENCSR474XFV	
ChIP-seq	H3K27ac	RWPE1	ENCSR203KEU	ENCFF708CBX	ENCFF939LTT
ChIP-seq	H3K27ac	SKNSH	ENCSR564IGJ	ENCFF380OTV	ENCFF959FMO
ChIP-seq	H3K27ac	Bipolar neuron	ENCSR905TYC	ENCFF751YAL	ENCFF687LIL
ChIP-seq	H3K27ac	GM23338	ENCSR729ENO	ENCFF403VXK	ENCFF754UFV
ChIP-seq	H3K27ac	C42B	ENCSR279KIX	ENCFF913EZV	ENCFF980IJT
ChIP-seq	H3K27ac	22Rv1	ENCSR391NPE	ENCFF025ZEN	ENCFF769UET
ChIP-seq	H3K27ac	Foreskin keratinocyte	ENCSR709ABP	ENCFF085FAH	ENCFF178GZR
ChIP-seq	H3K27ac	Foreskin keratinocyte	ENCSR709ABP	ENCFF776HMQ	ENCFF178GZR
ChIP-seq	H3K27ac	Epithelial cell of prostate	ENCSR910PDW	ENCFF382XYO	ENCFF213AZI
ChIP-seq	H3K27ac	RWPE2	ENCSR987PNT	ENCFF245ORL	ENCFF169DGZ

ChIP-seq	H3K4me3	SKNSH	ENCSR975GZA	ENCFF027SGQ	ENCFF959FMO
ChIP-seq	H3K4me3	SKNSH	ENCSR975GZA	ENCFF245RXP	ENCFF959FMO
ChIP-seq	H3K4me3	NCIH929	ENCSR082NQB	ENCFF417RNS	ENCFF446RUP
ChIP-seq	H3K4me3	NCIH929	ENCSR082NQB	ENCFF067LLV	ENCFF446RUP
ChIP-seq	H3K4me3	Bipolar neuron	ENCSR849YFO	ENCFF096QTT	ENCFF687LIL
ChIP-seq	H3K4me3	Bipolar neuron	ENCSR849YFO	ENCFF950QWN	ENCFF687LIL
ChIP-seq	H3K4me3	Muscle of leg	ENCSR128QKM	ENCFF552OGD	ENCFF622XBJ
ChIP-seq	H3K4me3	Heart right ventricle	ENCSR107RDP	ENCFF897OOT	ENCFF246SXV
ChIP-seq	H3K4me3	Gastrocnemius medialis	ENCSR098OLN	ENCFF310NMI	ENCFF587DDD
ChIP-seq	H3K4me3	OCILY3	ENCSR548PZS	ENCFF816RLY	ENCFF691EEI
ChIP-seq	NR2C1	GM12878	ENCSR784VIQ	ENCFF785FLS	ENCFF322NTO
ChIP-seq	EP300	Ovary	ENCSR696LQU	ENCFF405UYE	ENCFF271JKY
ChIP-seq	NFXL1	GM12878	ENCSR746XEG	ENCFF673BXM	ENCFF322NTO
ChIP-seq	MXI1	Neural cell	ENCSR934NHU	ENCFF260PNL	ENCFF056HWK
ChIP-seq	ZNF318	K562	ENCSR334HSW	ENCFF373YTD	ENCFF790TAN
ChIP-seq	CREB1	HepG2	ENCSR112ALD	ENCFF011HOS	ENCFF950AXC
ChIP-seq	CTCF	RWPE1	ENCSR303GFI	ENCFF204KRO	ENCFF290UZX
ChIP-seq	RFX1	MCF7	ENCSR788XNX	ENCFF804LEF	ENCFF426RDP
ChIP-seq	CTCF	Ascending aorta	ENCSR960MDF	ENCFF353ZVY	ENCFF023NJF
ChIP-seq	E4F1	K562	ENCSR731LHZ	ENCFF978NVP	ENCFF910IKB
DNase-seq	Open chromatin	Left arm bone	ENCSR976XOY	ENCFF205JXZ	
DNase-seq	Open chromatin	A673	ENCSR346JWH	ENCFF348KWA	
DNase-seq	Open chromatin	T-helper 1 cell	ENCSR000EQC	ENCFF425YMJ	
DNase-seq	Open chromatin	Retina	ENCSR820ICX	ENCFF441YDL	
DNase-seq	Open chromatin	Uterus	ENCSR129BZE	ENCFF759POB	
DNase-seq	Open chromatin	NAMALWA	ENCSR301OGM	ENCFF554YJG	
DNase-seq	Open chromatin	SKMEL5	ENCSR000FEK	ENCFF844BZM	
DNase-seq	Open chromatin	ELF1	ENCSR678ILN	ENCFF433CFI	
DNase-seq	Open chromatin	Myocyte	ENCSR000EPD	ENCFF042QTI	
DNase-seq	Open chromatin	Pancreas	ENCSR828FVZ	ENCFF984FKS	

Table 2. Datasets used from ENCODE as training data for LanceOtron's deep learning neural network.

Each BAM file was downloaded directly from ENCODE, along with the corresponding control BAMs for H3K27ac, H3K4me3, and TF ChIP-seq experiments. If multiple replicates of the control experiments existed, only the first listed in ENCODE's database was used for analysis. BAM files were sorted and indexed using Samtools 1.3 (`samtools sort filename.bam` and `samtools index filename.bam.sorted` commands respectively). Bigwig file coverage maps were created from the BAM files using the DeepTools 3.0.1 commands: `bamCoverage --bam filename.bam.sorted -o filename.bw --extendReads -bs 1 --normalizeUsing RPKM` for paired-end sequenced experiments. For single-end sequenced experiments the average fragment length was obtained from ENCODE and used with the --extendReads flag, making the command: `bamCoverage --bam filename.bam.sorted -o filename.bw --extendReads averageFragmentLength -bs 1 --normalizeUsing RPKM`.

Putative peak calls were carried out on all datasets, whereby regions would be verified as either peak or noise based on visual inspection. Coordinates for the regions being assessed were determined three ways. The MACS2 peak caller was used on default settings, `macs2 callpeak -t filename.bam.sorted -c control_filename.bam.sorted -n sample_label -f BAM -g hs -B -q 0.01` for H3K27ac, H3K4me3, and transcription factor ChIP-seq datasets. For ATAC and DNase, which lack control tracks, the following command was used: `macs2 callpeak -t filename.bam.sorted -n sample_label -f BAM -g hs -B -q 0.01`. The second and third peak call methods were based on labelling regions based on their fold enrichment compared to the mean signal. Coverage maps of sequenced reads were first smoothed by applying a rolling average of a given window size. If this smoothed signal was greater than the mean multiplied by a fold enrichment threshold, the coordinate was marked as enriched; adjacent enriched regions were then merged. Five different smoothing windows were used (100bp, 200bp, 400bp, 800bp, 1600bp) as well as five different enrichment thresholds (1, 2, 4, 8, 16). Method two compared the smoothed signal to the mean of chromosome-wide signal multiplied by fold enrichment. Method three was similar except the smoothed signal was compared to either the mean of the chromosome, surrounding 5kb, or surrounding 10kb, whichever value was highest (i.e. max[chromosome mean, 5kb mean, 10kb mean]).

From each dataset a 1Mb continuous region was selected at random for each chromosome for autosomes and sex chromosomes only. If the start of the randomly selected region was near the end of the chromosome, the area considered was from that point to the chromosome end, then from the chromosome start extending until a full 1MB was covered. Peaks called from all 3 methods which started within the random region were made available for labelling. For both of the mean-based methods, a peak call was made for each permutation of the smoothing window and enrichment threshold parameters, and all 25 calls were combined - this meant the presence of multiple overlapping candidate peaks in some cases. A python implementation of BEDTools[28] (pybedtools) was used to find overlapping peaks, and only one selected at random was considered for visual inspection.

Only candidate peaks which were obviously peaks or noise were labelled as such. Visual inspection was carried out using MLV[14], with control tracks overlaid when available. Regions were inspected one at a time, until 100 verified peaks were found for the dataset or all of the regions were assessed. Entire 1Mb regions were assessed (no early stopping), with the order of chromosomes randomized. A total of 736,753 regions were labelled this way (5,016 peaks and 731,737 noise regions) covering 499Mb.

The candidate peak selection algorithm was also called on these tracks (see below), and the regions overlapping with the hand labelled peaks were also included in the training data, resulting in an additional 3,447 regions. Noise regions were sampled down to match the number of peak regions (8,500 were selected). Prioritization was given to regions labelled noise with the highest signal, and all regions with a max height in the 25th percentile or greater were included (3,658) for training, with the remaining noise regions randomly sampled. Ultimately 16,963 regions were used for training: 8500 noise regions plus 8,463 peaks (ATAC-seq: 1,926; DNase-seq: 2,097; H3K27ac ChIP-seq: 1,651; H3K4me3 ChIP-seq: 1,806; transcription factor ChIP-seq: 983).

Wide and deep convolutional neural network to learn shape and enrichment of regions

LanceOtron's machine learning architecture is a type of wide and deep neural network, combining enrichment values, logistic regression, and a CNN. The logistic regression model took as inputs the enrichment values, while the CNN used the 2kb of signal centered on the region of interest. The outputs of these two models, along with the 11 enrichment values, were input into a multilayer perceptron, which output the final peak score.

The logistic regression model was trained separately with the same training data, and all coefficients and model parameters saved. The wide and deep model was trained with the logistic regression component locked, and with loss distributed 70:30 to wide-and-deep-output:CNN-only-output. By penalizing the model on the CNN separately, it actively encouraged predictions from the 2kb of signal, i.e. the shape of the peak, to be accurate in absence of enrichment information.

To determine the optimal structure and hyperparameters, a brute force method of building many models with different configurations was carried out. In total 5,000 models were trained and tested using the python package Keras Tuner, though performance was robust across a range of configurations (**Supplementary fig. 1**). Model performance was assessed by measuring the number of correctly predicted classifications of enriched regions from data unseen to the model. The top 10 performing models were then subjected to 5-fold cross validation, and the architecture from the top performer was used.

Candidate peak selection

To optimize resources, candidate peaks are selected for their enrichment, whereby signal is extracted and passed to LanceOtron's neural network. We developed an algorithm which acts as a loose filter, allowing even modestly enriched regions through, and also helps to center the area around the highest signal, improving model performance. First the raw signal is smoothed by calculating the rolling mean for the surrounding 400bp, and any coordinate where the signal is fold*mean-chromosome-signal (4-fold enrichment above mean initially) is marked as enriched. Adjacent enriched regions are combined, and if the size is between 50bp and 2kb it is considered a candidate peak. Regions smaller than 50bp are discarded, and regions above 2kb are recursively reevaluated at a fold higher threshold until the region size is between 50bp and 2kb, or the region is greater than 20-fold enriched.

Peak caller benchmarking

Labelling testing data and calculating model performance

Testing datasets were also obtained from ENCODE (**Table 3**), but were not used in LanceOtron's training data. Each track was downloaded as a BAM file, and converted to bigwig using the same deeptools commands given above for in training data preparation. Chromosomes were shuffled (mitochondrial and alternative mapping chromosomes were excluded), and 1Mb was labelled for peaks or noise; regions which were not clearly either were excluded. For CTCF, H3K27ac, and H3K4me3 ChIP-seq datasets, 10 chromosomes each were labelled in this manner, and for ATAC and DNase, three chromosomes each. True positives, false positives, true negatives, and false negatives were determined by intersecting peak calls from LanceOtron and MACS2 with these labelled data using BedTools. True positives were found by using the command `bedtools intersect -a peak_call.bed -b labelled_peaks.bed -u -wa`. False negatives used `bedtools intersect -a peak_call.bed -b labelled_peaks.bed -v -wa`. True negatives used the command `bedtools intersect -a peak_call.bed -b labelled_noise.bed -v -wa`, while false positives used `bedtools intersect -a peak_call.bed -b labelled_noise.bed -u -wa`.

Experiment type			ENCODE ID numbers		
Assay	Target	Tissue	Experiment	BAM file	Control BAM
ATAC-seq	Open chromatin	MCF-7	ENCSR422SUG	ENCFF346MIJ	
ChIP-seq	CTCF	Spleen	ENCSR692ILH	ENCFF903NKV	ENCFF376BTL
ChIP-seq	H3K27ac	HAP-1	ENCSR131DVD	ENCFF742SZS	ENCFF247DSQ
ChIP-seq	H3K4me3	MG63	ENCSR579SNM	ENCFF996ZSR	ENCFF381RWF
DNase-seq	Open chromatin	A549	ENCSR000ELW	ENCFF410CDT	

Table 3. Datasets used from ENCODE as testing data for benchmarking peak callers.

Motif analysis

A custom motif matching script was written to match CTCF sites using a simple Python regex function. The motif position weight matrix (PWM) was downloaded from JASPAR[29] and the genomic coordinates matching the motif (and reverse complement) were recorded as a bed file. The matching sequence had to be the same length, with all nucleotides present at 75% or higher in the PWM as exact matches. With the bed file of the motif coordinates made, we once again employed BEDTools to find intersections with the peak calls.

Bed files which were exclusively LanceOtron or MACS2, as well as the intersections with promoter or enhancer regions, and TSSs were also found using BEDTools. The bed files listing the coordinates of the promoters or enhancers were from GenoSTAN[27], and for TSSs we used RefTSS[30].

The heat map of the coverage was made using the deeptools command:

```
computeMatrix reference-point -S CTCF_spleen_ENCFF656CCY.bw -R CTCF-spleen_LoT-only-peaks.bed CTCF-spleen_MACS2-only-peaks.bed --referencePoint center -a 1000 -b 1000 -out CTCF-spleen_LoT-and-MACS2_matrix.tab.gz
```

Followed by the command:

```
plotProfile -m CTCF-spleen_LoT-and-MACS2_matrix.tab.gz -out CTCF-spleen_LoT-and-MACS2.png --samplesLabel "Peak caller exclusive regions" --regionsLabel "LanceOtron only" "MACS2 only" --plotType=heatmap
```

Code availability

Code for the deep learning model is available at https://github.com/LHentges/LanceOtron with the webtool found at https://github.com/Hughes-Genome-Group/mlv.

Acknowledgments The authors would like to thank Dr. Simon J. McGowan for web design insight, project discussions and proofreading, as well as Dr. Jon Kerry and Dr. Dominic Waithe for their pilot research. We also want to acknowledge the many beta testers and users during the software development, in particular efforts from the Hughes Lab, AVI Group, Higgs Group, Milne Group, and the Centre for Computational Biology at the MRC Weatherall Institute of Molecular Medicine, University of Oxford. This work was also supported by the National Institutes of Health (USA) (R24DK106766 to J.R.H.), the Medical Research Council (MC_UU_12025 to S.T. and MC_UU_00016/14 to J.R.H.) and a Wellcome Trust Strategic Award (106130/Z/14/Z to J.R.H.).

Author contributions J.R.H. and S.T. designed the project and directed the research. L.D.H. built the candidate peak calling algorithm, labelled the training data, coded, trained, and tested the deep learning model, and created the command line tool. M.J.S. designed and coded the graphical user interface as well as the interactive visualization tools and built the website. D.J.D., J.R.H., and S.T. tested the software and suggested new features.

Supplementary Materials

Functionality of LanceOtron's user interface

LanceOtron features a rich graphical user interface, accessible using any web browser, and allows peak calls to be made without the use of the command line. Using the web tool to perform a peak call is demonstrated in **supplementary video 1**: https://youtu.be/k8GrIp55vDg. Furthermore, exploring and filtering data is also easily carried out with the graphical interface (**supplementary video 2**: https://youtu.be/M5ox8XI-U4Q).

1. Klein, D. C. & Hainer, S. J. Genomic methods in profiling DNA accessibility and factor

localization. *Chromosome Res.* **28**, 69–85 (2020).

2. Park, P. J. ChIP–seq: advantages and challenges of a maturing technology. *Nature Reviews Genetics* vol. 10 669–680 (2009).
3. Amemiya, H. M., Kundaje, A. & Boyle, A. P. The ENCODE Blacklist: Identification of Problematic Regions of the Genome. *Sci. Rep.* **9**, 9354 (2019).
4. Thomas, R., Thomas, S., Holloway, A. K. & Pollard, K. S. Features that define the best ChIP-seq peak calling algorithms. *Brief. Bioinform.* **18**, 441–450 (2017).
5. Zhang, Y. *et al.* Model-based analysis of ChIP-Seq (MACS). *Genome Biol.* **9**, R137 (2008).
6. Wilbanks, E. G. & Facciotti, M. T. Evaluation of algorithm performance in ChIP-seq peak detection. *PLoS One* **5**, e11471 (2010).
7. Auerbach, R. K. *et al.* Mapping accessible chromatin regions using Sono-Seq. *Proc. Natl. Acad. Sci. U. S. A.* **106**, 14926–14931 (2009).
8. Vega, V. B., Cheung, E., Palanisamy, N. & Sung, W.-K. Inherent signals in sequencing-based Chromatin-ImmunoPrecipitation control libraries. *PLoS One* **4**, e5241 (2009).
9. Hocking, T. D. *et al.* Optimizing ChIP-seq peak detectors using visual labels and supervised machine learning. *Bioinformatics* **33**, 491–499 (2017).
10. Stanton, K. P., Jin, J., Lederman, R. R., Weissman, S. M. & Kluger, Y. Ritornello: high fidelity control-free chromatin immunoprecipitation peak calling. *Nucleic Acids Res.* **45**, e173 (2017).
11. ENCODE Project Consortium. An integrated encyclopedia of DNA elements in the human genome. *Nature* **489**, 57–74 (2012).
12. Landt, S. G. *et al.* ChIP-seq guidelines and practices of the ENCODE and modENCODE consortia. *Genome Res.* **22**, 1813–1831 (2012).
13. Li, Q., Brown, J. B., Huang, H. & Bickel, P. J. Measuring reproducibility of high-throughput experiments. *The Annals of Applied Statistics* vol. 5 1752–1779 (2011).
14. Sergeant, M. J. *et al.* Multi Locus View : An Extensible Web Based Tool for the Analysis

of Genomic Data. doi:10.1101/2020.06.15.151837.

15. Baker, M. 1,500 scientists lift the lid on reproducibility. http://www.nature.com/news/1-500-scientists-lift-the-lid-on-reproducibility-1.19970 (2016) doi:10.1038/533452a.
16. Smith, D. The Garden of Forking Paths: the Hidden Statistical Consequences of Data Contingency and Researcher Degrees of Freedom in Cyclostratigraphic Analysis, and Why Most Published Results are False. doi:10.1002/essoar.10500564.1.
17. Kent, W. J. The Human Genome Browser at UCSC. *Genome Research* vol. 12 996–1006 (2002).
18. Robinson, J. T. *et al.* Integrative genomics viewer. *Nature Biotechnology* vol. 29 24–26 (2011).
19. Rye, M. B., Sætrom, P. & Drabløs, F. A manually curated ChIP-seq benchmark demonstrates room for improvement in current peak-finder programs. *Nucleic Acids Research* vol. 39 e25–e25 (2011).
20. LeCun, Y., Bengio, Y. & Hinton, G. Deep learning. *Nature* **521**, 436–444 (2015).
21. Wainberg, M., Merico, D., Delong, A. & Frey, B. J. Deep learning in biomedicine. *Nat. Biotechnol.* **36**, 829–838 (2018).
22. Zhou, J. & Troyanskaya, O. G. Predicting effects of noncoding variants with deep learning–based sequence model. *Nat. Methods* **12**, 931–934 (2015).
23. Kelley, D. R., Snoek, J. & Rinn, J. L. Basset: learning the regulatory code of the accessible genome with deep convolutional neural networks. *Genome Res.* **26**, 990–999 (2016).
24. Oh, D. *et al.* CNN-Peaks: ChIP-Seq peak detection pipeline using convolutional neural networks that imitate human visual inspection. *Sci. Rep.* **10**, 7933 (2020).
25. Cheng, H.-T. *et al.* Wide & Deep Learning for Recommender Systems. *Proceedings of the 1st Workshop on Deep Learning for Recommender Systems - DLRS 2016* (2016) doi:10.1145/2988450.2988454.
26. Tarbell, E. D. & Liu, T. HMMRATAC: a Hidden Markov ModeleR for ATAC-seq. *Nucleic*

Acids Res. **47**, e91 (2019).

27. Zacher, B. *et al.* Accurate Promoter and Enhancer Identification in 127 ENCODE and Roadmap Epigenomics Cell Types and Tissues by GenoSTAN. *PLoS One* **12**, e0169249 (2017).
28. Quinlan, A. R. & Hall, I. M. BEDTools: a flexible suite of utilities for comparing genomic features. *Bioinformatics* **26**, 841–842 (2010).
29. Fornes, O. *et al.* JASPAR 2020: update of the open-access database of transcription factor binding profiles. *Nucleic Acids Res.* **48**, D87–D92 (2020).
30. Abugessaisa, I. *et al.* refTSS: A Reference Data Set for Human and Mouse Transcription Start Sites. *J. Mol. Biol.* **431**, 2407–2422 (2019).